HRM Data Collection

Data access table entry

Thu, Dec 12, 2002

Using the HRM 10KHz digitizer for ordinary analog channels implies that a Data Access Table entry should be defined for copying the latest 10KHz data into the 15 Hz data pool. This is analogous to the `0x28`-type entry used for the IRM 1KHz digitizer.

There should not be any initialization/calibration required, as was done for the IRM, so the job is even easier. The current block number is read from a register, reduced by 1, and used to find the most recent complete set of 64 data words digitized within the last 100 μs. (It is necessary to decrement the block number because the current block number is the one into which data words are now being transferred. The previous block is complete.) These data words are then copied into the local data pool.

Consider the following layout of a new `RDATA` entry:

```
3200   chan   regAddress
bufAddress    0000   count
```

The `0x32` is the new type# that identifies this format. The `regAddress` would be the address of the register containing the current block number. The `bufAddress` would be the base address of the 2 MB circular buffer.

One could have the code obtain the two addresses from a call to `CINFOentry`, specifying entry type#5. In order not to make this call every 15 Hz cycle, though, there should be some context, and the only obvious context is the unused field of this `RDATA` entry.

To keep things simple, assume we just implement it the simple way. The user who installs the new entry should take care that all fields are correct. Note that in the case that there is more than one slow data module in an HRM, there will be more than one of these entries. The low bits of the `regAddress` will indicate which it is, as the `regAddress` is the address of the current slow data block number, not the base addresss of all the registers in that PMC board.

There is an assumption made that the channel number used for representing the first channel of a slow data A/D starts on a 64-channel boundary, so that the least significant 6 bits of the system channel number specify the hardware channel#. But there is no assumption made that the first channel number of a block be `0x0100` or `0x0140`, as it was in the case of the 68K-based IRM code; rather, it merely needs to be a multiple of `0x40`.

An example of what might commonly be found in a real system is as follows:

```
3200 0100 5050 0000
5000 0000 0000 0040
```

The register containing the current slow data block number is at `0x50500000`. The circular buffer base address is `0x50000000`. This entry copies the most recent 64 data words from the circular buffer and installs them into consecutive channels starting at channel `0x0100`.